

# HTML5



# My ID (HTML5, 1<sup>st</sup> semester 2020)

ID	성명
HM01	강서현
HM02	강태민
HM03	김세은
HM04	김준호
HM05	남승현
HM06	박종혁
HM07	심준혁
HM08	여수민
HM09	이상혁
HM10	정영훈

HM11	차혁준
HM12	최재형
HM13	하태헌
HM14	강권모
HM15	김민욱
HM16	김민준
HM17	김현식
HM18	신종원
HM19	장성운
HM20	조재윤
HM21	경명아

# Weekly plan (HTML5, 1<sup>st</sup> semester 2020)

- **wk01** : Introduction to curriculum, current state of HTML5 & github
- **wk02** : Making HTML5 documents
- **wk03** : iFrame, Media, my 1<sup>st</sup> Homepage & Intro to Semantic tags
- **wk04** : Semantic tags & Web forms
- **wk05** : **CSS3 I. Basic & Box model**
- **wk06** : **CSS3 II. Advanced & Animation**
- **wk07** : **my 2<sup>nd</sup> Homepage (hmn\_rpt02)**
- **wk08** : Mid-term Exam.
- **wk09** : **JS I. Data types & operators**
- **wk10** : **JS II.**
- **wk11** : **JS III.**
- **wk12** : **JS IV.**
- **wk13** : **JS V.**
- **wk14** : **my final Homepage (hmn\_rpt03)**
- **wk15** : Final exam.

## 간단한 계산기

첫번째 정수:

두번째 정수:

계산 결과 :

# 예비중간고사 안내

4

**일시 : 2020. 4. 30 (목) 6 교시(오후 2시) ~**

**장소 : 팀즈 실시간 강의**

**시험범위 : HTML5 & CSS3 (교재 1장 ~ 5장) --- 15점 (15%)**

- Project-2 (hmxx\_card.html) 평가
- HTML, CSS3에 대한 Kahoot Quiz 15개
- 중간고사 프로젝트 ( hmxx\_rpt02.html) 안내

**평가 기준: kahoot 배점 기준**

(1차 + 1차(speed) + 2차) 점수를 15점으로 환산

# [중간고사] Project-3. hmxx\_rpt02.html

일시 : 2020. 5. 1 ~ 5. 9 (토)

장소 : 집에서 준비하여 github에 업로드

내용 : **My Homepage V2 (배점: 15%)**

① **Semantic tag로 화면 layout 구성**

② 타이틀 : hmxx : 나는 누구인가?

③ 나의 하루

④ My Best 3 (책, 영화, 음악, 그림, 여행 등에서 선택 1) – datalist 사용

⑤ 동호회 모집 form.html 추가 (Web form, fieldset tag는 반드시 사용)

⑥ **X-MAS파티 초대장 (기 제작된 X-MAS 카드 활용)**

⑦ 적절한 그림, 음악, 동영상은 가능한 link로 처리해서 연결하고, 필요 시 media folder에 저장해서 이용.

→ hmxx\_rpt02 폴더를 github에 만든다.

→ hmxx\_rpt02 폴더에 README.md 파일을 만들어서 홈페이지에 대한 설명을 추가한다.

→ hmxx\_rpt02 폴더에 나의 홈페이지에 사용되는 모든 파일들을 올린다.

→ **나의 홈페이지 시작 파일은 hmxx\_rpt02.html**

**\* 제출기한 : 5월 9일(토) 밤 12시.**

# 06

자바스크립트 언어

# 강의 목표

1. 자바스크립트 언어의 요소와 구조를 이해한다.
2. 자바스크립트 코드를 웹 페이지에 삽입하는 방법을 안다.
3. 자바스크립트로 브라우저에 출력하고 사용자 입력 받는 방법을 안다.
4. 자바스크립트에서 다루는 데이터 타입과 변수에 대해 이해한다.
5. 자바스크립트의 연산자의 종류를 알고 사용할 수 있다.
6. 자바스크립트의 조건문의 종류를 알고 사용할 수 있다.
7. 자바스크립트의 반복문의 종류를 알고 사용할 수 있다.
8. 자바스크립트 함수를 작성할 수 있다.
9. 사용자에게 제공되는 eval(), parseInt() 등 자바스크립트 함수를 활용할 수 있다.

→ Easy calculator 제작



1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages





# 자바스크립트 Java script

JavaScript is the most popular programming language in the world.

It is the language for HTML, for the Web, for computers, servers, laptops, tablets, smart phones, **IoT**, **AI** and more.

**tensorflow.js**

HTML and CSS

Learn HTML

Learn CSS

Learn W3.CSS

Learn Colors

Learn Bootstrap 3

Learn Bootstrap 4

Learn Icons

Learn Graphics

Learn How To

JavaScript

Learn JavaScript

Learn jQuery

Learn AngularJS

Learn JSON

Learn AJAX

Server Side

Learn SQL

Learn PHP

Learn ASP

Learn Node.js

Learn Raspberry Pi

# 자바스크립트 확장

10



JavaScript Learning Curves



jQuery



Node.js



AngularJS

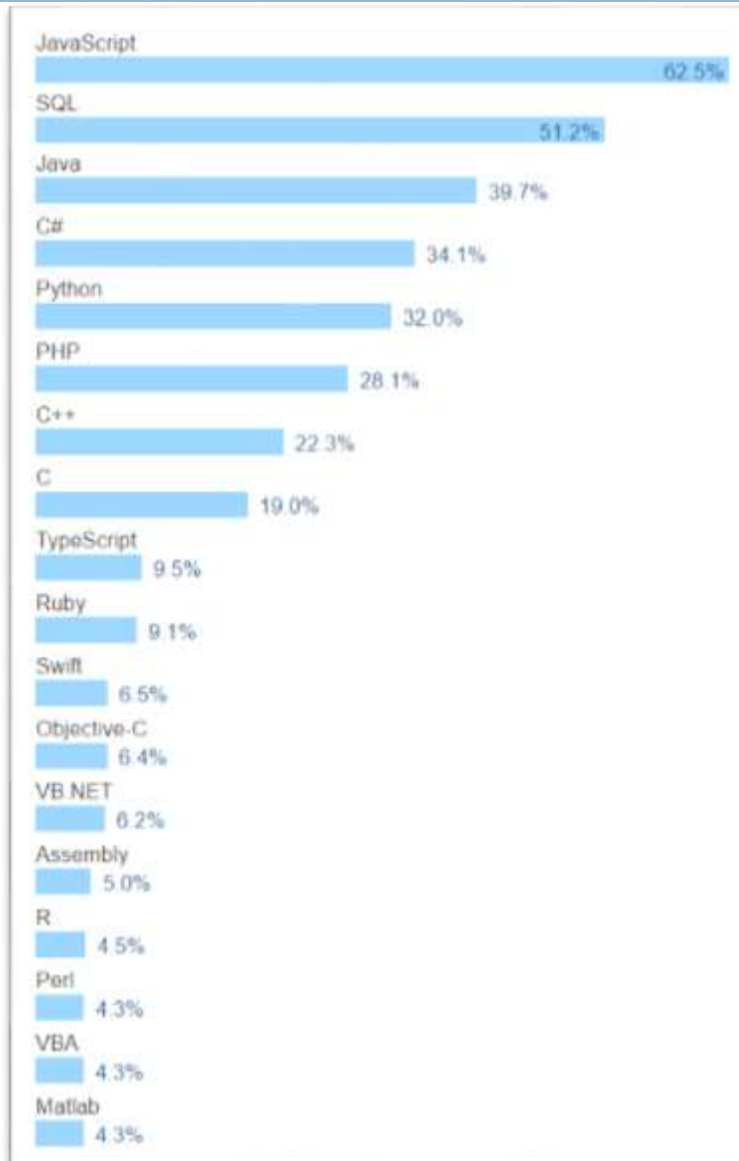
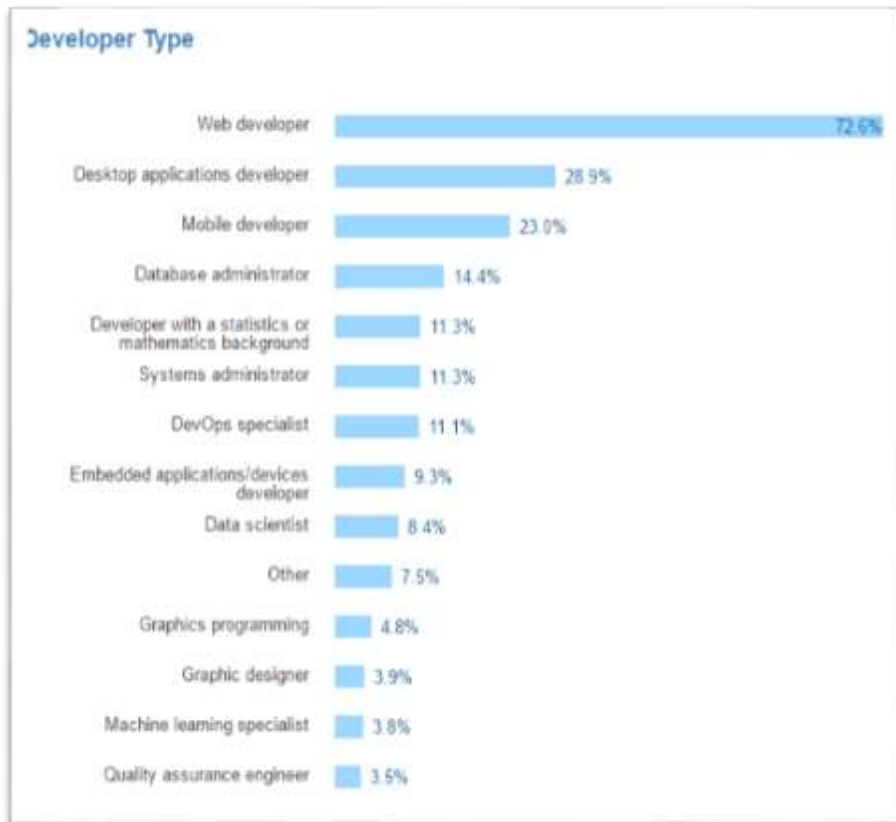


# 자바스크립트 현황

<https://stackoverflow.com/insights/survey/2017>

11

Favorite techniques



# 자바스크립트 언어

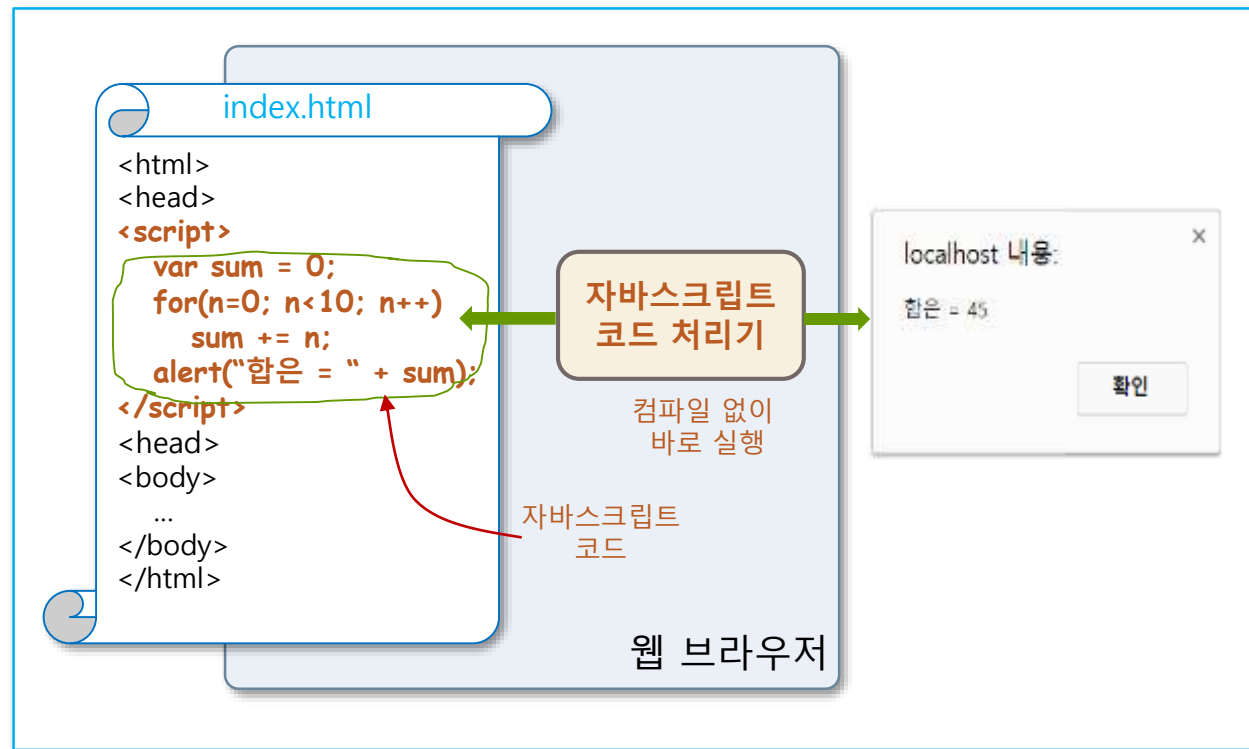
12

## □ Javascript

- 1995년 넷스케이프사의 브렌던 아이크(**Brendan Eich**)가 개발
- Netscape Navigator 2.0 브라우저에 최초 탑재
- 웹 프로그래밍 개념 창시

## □ 특징

- HTML 문서에 내장
  - 조각 소스 코드
- 스크립트 언어
  - 인터프리터 실행
  - 컴파일 필요 없음
- 단순
  - C언어 구조 차용
  - 배우기 쉬움



# 웹 페이지에서 자바스크립트의 역할

13

- 사용자의 입력 및 계산
  - ▣ 마우스와 키보드 입력은 오직 자바스크립트로만 가능
  - ▣ 계산 기능
- 웹 페이지 내용 및 모양의 동적 제어 (**DOM**)
  - ▣ **HTML 태그의 속성, 콘텐츠, CSS 프로퍼티 값 동적 변경**
- 브라우저 제어 (**BOM**)
  - ▣ 브라우저 윈도우 크기와 모양 제어
  - ▣ 새 윈도우 열기/닫기
  - ▣ 다른 웹 사이트 접속
  - ▣ 히스토리 제어
- 웹 서버와의 통신
- 웹 애플리케이션 작성
  - ▣ 캔버스 그래픽, 로컬/세션 스토리지 저장, 위치정보서비스 등
  - ▣ **IoT dash board**
  - ▣ **Mobile AI engine, ...**

# 자바스크립트 코드의 위치

14

- 자바스크립트 코드 작성 가능한 위치
  1. HTML 태그의 이벤트 리스너 속성에 작성 (**inline JS**)
  2. **<script></script>** 태그에 작성
  3. 자바스크립트 파일에 작성 (\*.js)
  4. URL 부분에 작성 **<a href="js\_code">**

## 1. HTML 태그의 이벤트 리스너에 자바스크립트 코드 작성

onclick 이벤트 리스너 속성      자바스크립트 코드 (이미지를 banana.png로 교체)

```

```

↑

# 예제 6-1 HTML 태그의 이벤트 리스너 속성에 자바스크립트 코드 작성

15

```
<!DOCTYPE html>
<html>
<head>
<title>이벤트 리스너 속성에 자바스크립트 코드</title>
</head>
<body>
<h3>마우스 올려 보세요</h3>
<hr>

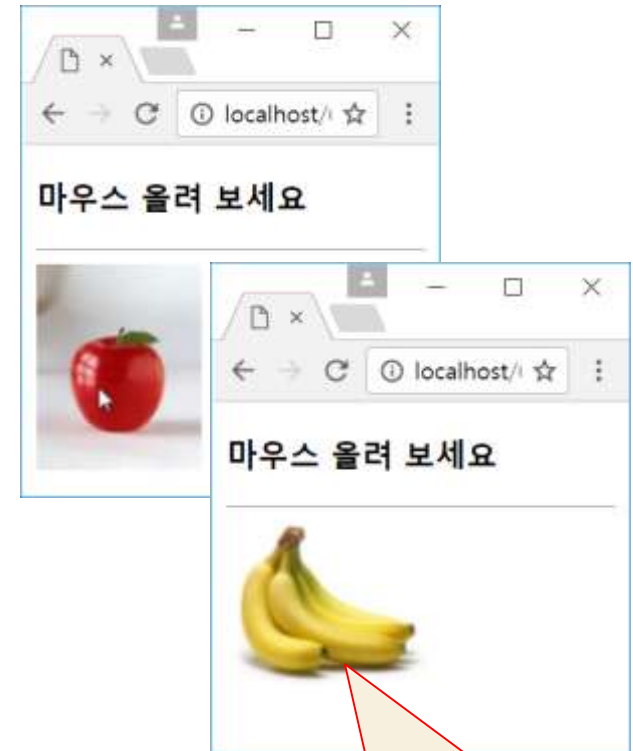
</html>
```

이벤트 리스너  
속성

this는 현재 img 태그를  
가리키는 자바스크립트 키워드

자바스크립트  
코드

wk09js\_ex01\_inline.html



이미지에 마우스를 올리면 바나나로  
내리면 다시 사과로 바뀐다.



# <script> </script> 태그에 자바스크립트 작성

16

## □ 특징

- ▣ <head> </head>나 <body> </body> 내 어디든 가능
- ▣ 웹 페이지 내에 여러 번 삽입 가능

# 예제 6-2 <script> 태그에 자바스크립트 코드 작성

17

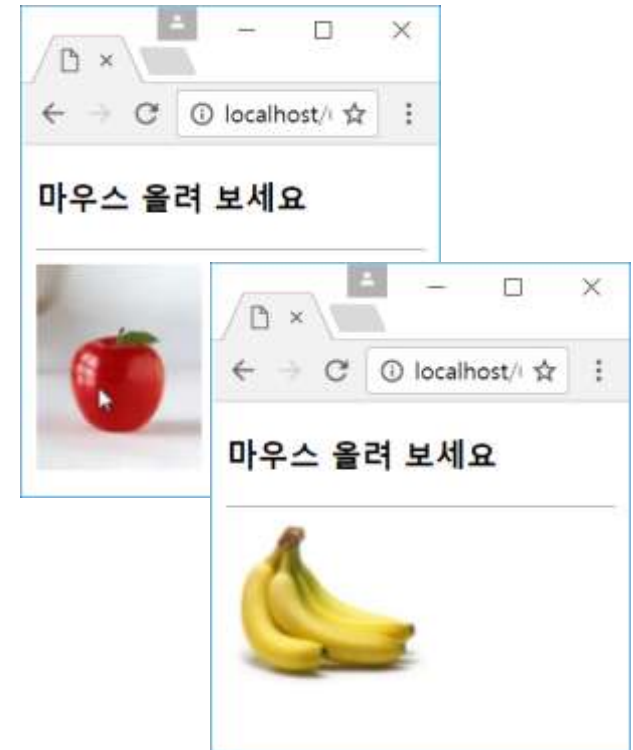
```
<!DOCTYPE html>
<html>
<head> <title>script 태그에 자바스크립트 작성</title>
<script>
function over(obj) {
    obj.src="media/banana.png";
}
function out(obj) {
    obj.src="media/apple.png";
}
</script>
</head>
<body>
<h3>마우스 올려 보세요</h3>
<hr>

</body>
</html>
```

obj는 전달받은  
img 태그를 가리킴

this는 현재 img 태그를  
가리키는 자바스크립트 키워드

wk09js\_ex02\_script\_tag.html



# 자바스크립트 코드를 별도 파일에 작성

18

- 자바스크립트 코드 파일 저장
  - ▣ 확장자 **.js** 파일에 저장
  - ▣ `<script>` 태그 없이 자바스크립트 코드만 저장
- 여러 웹 페이지에서 불러 사용
  - 웹 페이지마다 자바스크립트 코드 작성 중복 불필요
  - `<script>` 태그의 `src` 속성으로 파일을 불러 사용

```
<script src="파일이름.js">  
    // 이곳에 자바스크립트 코드 추가 작성 가능  
</script>
```

# 예제 6-3 자바스크립트 파일 작성 및 불러오기

19

예제 6-2의 <script> 태그에 들어 있는 자바스크립트 코드를 lib.js 파일에 저장하고 불러와서 사용하도록 수정하라.

```
/* 자바스크립트 파일 lib.js */  
function over(obj) {  
    obj.src="media/banana.png";  
}  
function out(obj) {  
    obj.src="media/apple.png";  
}
```

lib.js

lib.js  
불러오기

```
<!DOCTYPE html>  
<html>  
<head><title>외부 파일에 자바스크립트 작성</title>  
<script src="lib.js">  
</script>  
</head>  
<body>  
<h3>마우스 올려 보세요</h3>  
<hr>  
  
</body>  
</html>
```

마우스 올려 보세요 (using js file)



마우스 올려 보세요 (using js file)

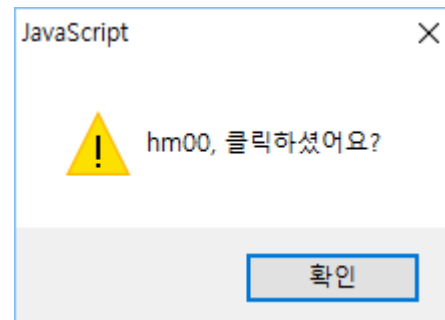
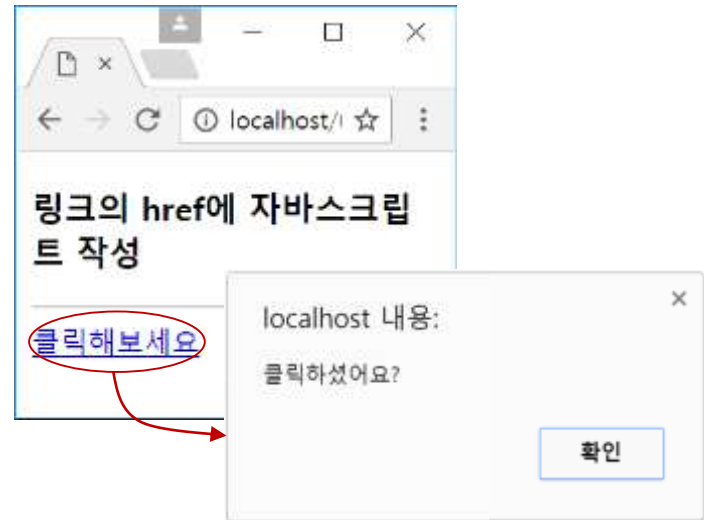


# 예제 6-4 링크의 href에 자바스크립트 코드 작성

20

```
<!DOCTYPE html>
<html>
<head> <title>URL에 자바스크립트 작성</title>
</head>
<body>
<h3>링크의 href에 자바스크립트 작성</h3>
<hr>
<a href="javascript:alert('클릭하셨어요?')">
  클릭해보세요</a>
</body>
</html>
```

wk09js\_ex04\_script\_href.html



# 자바스크립트로 HTML 콘텐츠 출력

21

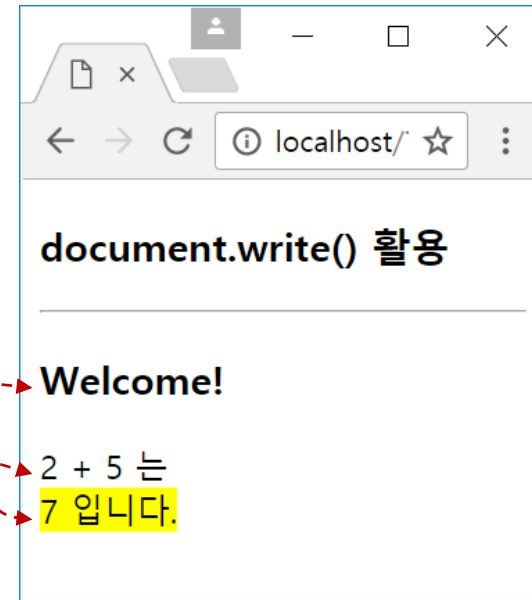
- 자바스크립트로 HTML 콘텐츠를 웹 페이지에 직접 삽입
  - ▣ 바로 브라우저 윈도우에 출력
  - ▣ **document.write()**
    - 예) `document.write("<h3>Welcome!</h3>");`
  - ▣ **document.writeln()**
    - `writeln()`은 텍스트에 'Wn' 을 덧붙여 출력
    - 'Wn'을 덧붙이는 것은 고작해야 빈칸 하나 출력
    - 다음 줄로 넘어가는 것은 아님

[https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref\\_doc\\_writeln](https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_doc_writeln)

# 예제 6-5 document.write()로 웹 페이지에 HTML 콘텐츠 출력

22

```
<!DOCTYPE html>
<html>
<head> <title>document.write() 활용</title>
</head>
<body>
<h3>document.write() 활용</h3>
<hr>
<script>
  document.write("<h3>Welcome!</h3>");
  document.write("2 + 5 는 <br>");
  document.write("<mark>7 입니다.</mark>");
</script>
</body>
</html>
```



wk09js\_ex05\_document\_write.html



# 자바스크립트 다이얼로그

localhost 내용: ×

이름을 입력하세요

확인

취소

localhost 내용: ×

클릭하였습니다.

확인

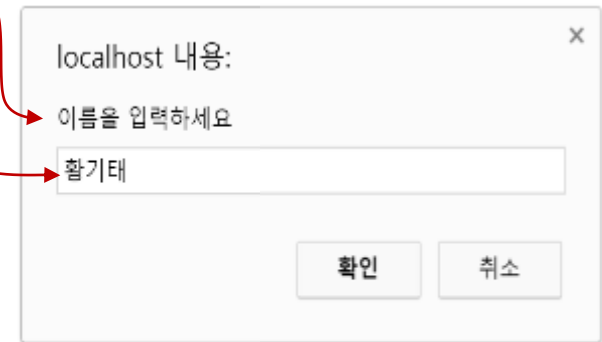
# 자바스크립트 다이얼로그 : 프롬프트 다이얼로그

24

- **prompt("메시지", "디폴트 입력값")** 함수
  - 사용자로부터 문자열을 입력 받아 리턴

```
var ret = prompt("이름을 입력하세요", "황기태");  
if(ret == null) {  
    // 취소 버튼이나 다이얼로그를 닫은 경우  
}  
else if(ret == "") {  
    // 문자열 입력 없이 확인 버튼 누른 경우  
}  
else {  
    // ret에는 사용자가 입력한 문자열  
    document.write("입력한 이름은 " + ret + " 입니다.");  
}
```

wk09js\_ex05dialog\_prompt.html



JS dialog : prompt()

입력한 이름은 황기태 입니다.

# 자바스크립트 다이얼로그 : 확인 다이얼로그

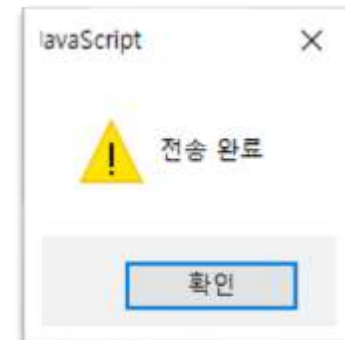
25

## ■ **confirm**("메시지") 함수

- “메시지”를 출력하고 ‘확인/최소(OK/CANCEL)’버튼을 가진 다이얼로그 출력
- ‘확인’ 버튼을 누르면 true, '취소' 버튼이나 강제로 다이얼로그를 닫으면 false 리턴

```
var ret = confirm("전송할까요");  
if(ret == true) {  
    // 사용자가 "확인" 버튼을 누른 경우  
    alert("전송 완료");  
}  
else {  
    // 취소 버튼이나 다이얼로그를 닫은 경우  
}
```

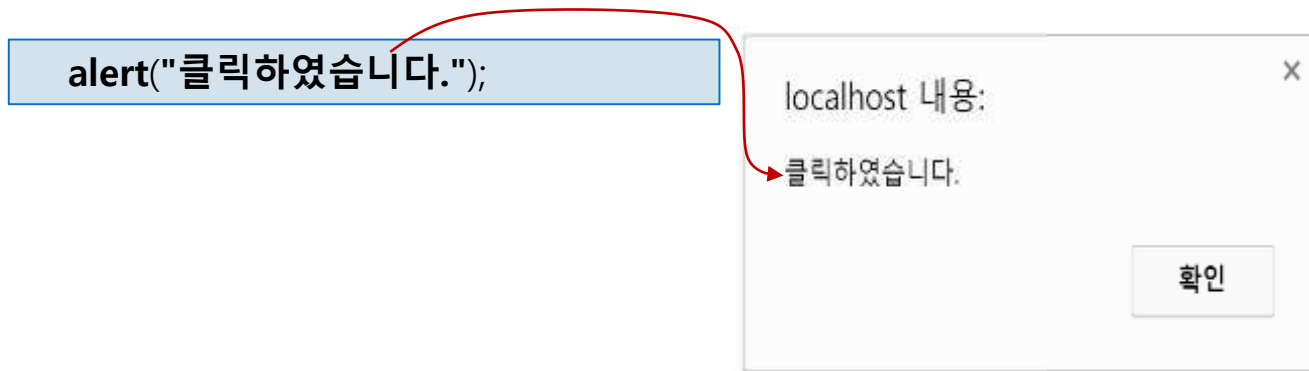
wk09js\_ex05dialog2\_confirm.html



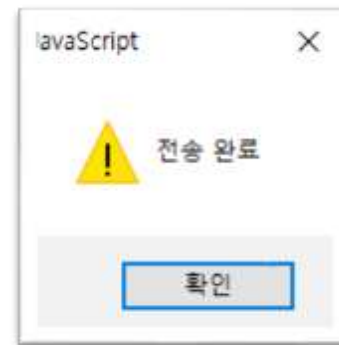
# 자바스크립트 다이얼로그 : 경고 다이얼로그

26

- **alert("메시지")** 함수 → modal window
  - ▣ 메시지'와 '확인' 버튼을 가진 다이얼로그 출력, 메시지 전달



- ▣ 확인 다이얼로그에서 호출



# 자바스크립트 식별자

변수,

상수(리터럴),

함수

# 자바스크립트 식별자 (exam)

28

## □ 식별자

▣ 자바스크립트 프로그램의 변수, 상수(리터럴), 함수의 이름

▣ 식별자 만드는 규칙

- 첫 번째 문자 : 알파벳(A-Z, a-z), 언더스코어(\_), \$ 문자만 사용 가능
- 두 번째 이상 문자 : 알파벳, 언더스코어(\_), 0-9, \$ 사용 가능
- 대소문자는 구분되어 다루어짐
  - *myHome*과 *myhome*은 다른 식별자
- 자바스크립트 예약어 사용 불가
  - *false, for, if, null* 등 자바스크립트 예약어 사용 불가

▣ 식별자 사용 사례

```
6variable;      // (x) 숫자로 시작할 수 없음
student_ID;     // (0)
_code;          // (0) 맞지만 권하지 않음
if;             // (x) 예약어 if 사용 불가
%calc           // (x) % 사용 불가
bar, Bar;       // (0) bar와 Bar는 서로 다른 식별자임에 주의
```

# 자바스크립트 문장 (coding 방법)

29

## □ 문장

- 자바스크립트 프로그램의 기본 단위는
- 문장과 문장을 구분하기 위해 세미콜론(;) 사용

```
i = i + 1           // (0) 한 줄에 한 문장만 있는 경우 세미콜론 생략 가능  
j = j + 1;          // (0)  
k = k + 1; m = m + 1; // (0) 한 줄에 여러 문장  
n = n + 1; p = p + 1; // (x) 첫 번째 문장 끝에 세미콜론이 필요함
```

## □ 주석문

```
// 한 라인 주석. 라인의 끝까지 주석 처리  
/*  
    여러 라인 주석  
*/
```



# 자바스크립트 데이터 타입 **data types**

숫자

논리

문자열

객체

**null**

# 데이터 타입 (types of data)

31

- 자바스크립트 언어에서 다루는 데이터 종류
  - ▣ 숫자 타입 : 정수, 실수(예: 42, 3.14)
  - ▣ 논리 타입 : 참, 거짓(예: **true**, **false**)
  - ▣ 문자열 타입(예: '좋은 세상', "a", "365", "2+4")
  - ▣ 객체 레퍼런스 타입 : 객체를 가리킴. C 언어의 포인터와 유사
  - ▣ **null** : 값이 없음을 표시하는 특수 키워드. Null, NULL과는 다름
  
- 특징
  - ▣ 자바스크립트에는 **문자 타입 없음**. 문자열로 표현

# 자바스크립트 변수

## variable

```
var score;                // 변수 score 선언
var year, month, day;     // year, month, day의 3 개의 변수 선언
var address = "서울시";
var k = 7;
```

**dynamic typing !!!**  
**→ 동적 데이터형결정**

# 변수 (variable)

33

- 변수 : 자바스크립트 데이터 저장 공간
  - ▣ 변수 선언 : 변수 이름을 정하고, 저장 공간 할당
    - var 키워드로 선언하는 방법

```
var score;           // 변수 score 선언
var year, month, day; // year, month, day의 3 개의 변수 선언
var address = "서울시"; // address 변수를 선언하고 "서울시"로 초기화
```

- var 없이 선언

```
age = 21;           // var 없이, 변수 age를 선언하고 21로 초기화
```

- *age*가 이미 선언된 변수이면, 존재하는 *age*에 21 저장

- ▣ 자바스크립트에는 변수 타입 없음
  - 세부 변수 타입 선언하지 않음 (dynamic typing)

```
var score; // 정상적인 변수 선언
int score; // 오류. 변수 타입 int 없음
```

- 변수에 저장되는 값에 대한 제약 없음

```
score = 66.8; // 실수도 저장 가능
score = "high"; // 문자열로 저장 가능
```

# 지역변수와 전역변수

34

지역변수 (local variable)	전역변수 (global variable)
함수 내에 var 키워드로 선언	함수 밖에 선언되거나, 함수 내에 var 키워드 없이 선언된 변수
선언된 함수 내에서만 사용	프로그램 전역에서 사용

```
var x;    // 전역변수 x
function f() {
  var y;   // 지역변수 y 선언
  x = 10;  // 전역 변수 x에 10 저장
  y = 10;  // 지역 변수 y에 10 저장
  z = 10;  // 새로운 전역변수 z 선언. 10으로 초기화
}
```

- **let**
  - 함수 내부, for-loop 블록 안에서만 정의되는 **블록변수** 선언
- **const** : 상수 정의
  - const PI = 3.14;

# this로 전역변수 접근 (중요, exam)

35

- 지역 변수와 전역 변수의 이름을 같을 때
  - ▣ 전역 변수에 접근하고자 할 때 : **this.전역변수**

```
var x;    // 전역변수
function f() {
  var x;   // 지역변수
  x = 1;   // 지역변수 x에 1 저장
  this.x = 100; // 전역변수 x에 100 저장
}
```

## 예제 6-6 지역변수와 전역변수

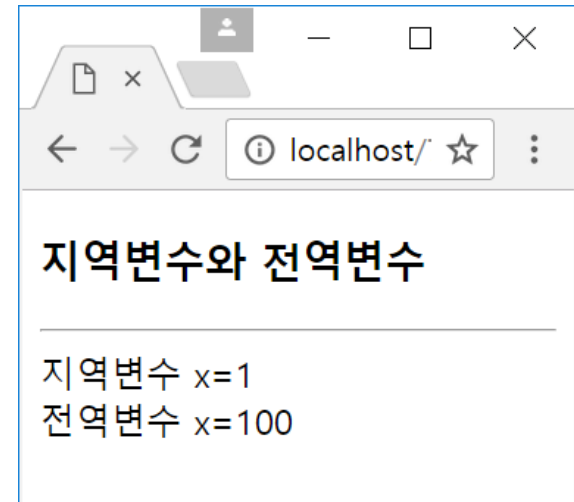
36

```
<!DOCTYPE html>
<html>
<head>
<title>지역변수와 전역변수</title> </head>
<body>
<h3>지역변수와 전역변수</h3>
<hr>
<script>
var x=100; // 전역변수 x

function f() { // 함수 f() 선언
    var x=1; // 지역변수 x

    document.write("지역변수 x=" + x);
    document.write("<br>");
    document.write("전역변수 x=" + this.x);
}

f(); // 함수 f() 호출
</script>
</body>
</html>
```





자바스크립트

상수

**literal**

# 자바스크립트의 상수 (exam)

38

- 상수(literal)
  - 변수에 지정되는 데이터 값 그 자체
- 상수 종류

상수의 종류		특징	예
정수	8진수	0으로 시작	var n = 015; // 8진수 15, 10진수로 13
	10진수		var n = 15; // 10진수 15
	16진수	0x로 시작	var n = 0x15; // 16진수 15, 10진수로 21
실수	소수형		var height = 0.1234;
	지수형		var height = 1234E-4; // $1234 \times 10^{-4} = 0.1234$
논리	참	true	var condition = true;
	거짓	false	var condition = false;
문자열		""로 묶음	var hello = "안녕하세요";
		''로 묶음	var name = 'kitae';
기타	null	값이 없음을 뜻함	var ret = null;
	NaN	수가 아님을 뜻함	var n = parseInt("abc"); // 이때 parseInt()는 NaN을 리턴

# 문자열 상수

39

- ▣ 이중 인용 부호(" ")와 단일 인용 부호(' ') 모두 사용
- ▣ 문자열 내에 문자열

`<p onmouseover = "document.body.style.color = 'brown' ">`

문자열 내 문자열

- “ 문자를 그대로 사용하고자 하는 경우 **₩**”로 사용할 것

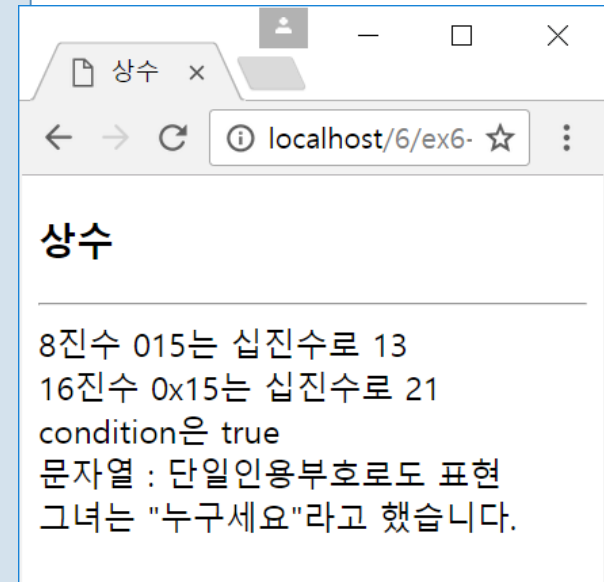
```
var cite="그녀는 ₩누구세요₩"라고 했습니다.;
```

# 예제 6-7 상수

40

```
<!DOCTYPE html>
<html>
<head> <title>상수</title> </head>
<body>
<h3>상수</h3>
<hr>
<script>
  var oct = 015; // 015는 8진수. 10진수로 13
  var hex = 0x15; // 0x15는 16진수. 10진수로 21
  var condition = true; // True로 하면 안됨

  document.write("8진수 015는 십진수로 " + oct + "<br>");
  document.write("16진수 0x15는 십진수로 " + hex + "<br>");
  document.write("condition은 " + condition + "<br>");
  document.write("문자열 : 단일인용부호로도 표현" + "<br>");
  document.write("그녀는 W"누구세요W"라고 했습니다.");
</script>
</body>
</html>
```



wk09js\_ex07\_literal.html

# 자바스크립트 식과 연산

연산 종류	연산자	연산 종류	연산자
산술	+ - * / %	대입	= *= /= += -= &= ^=  = <<= >>= >>>=
증감	++ --	비교	> < >= <= == !=
비트	&   ^ ~	논리	&&    !
시프트	>> << >>>	조건	? :

# 자바스크립트의 식과 연산

42

## 자바스크립트의 연산과 연산자 종류

연산 종류	연산자	연산 종류	연산자
산술	+ - * / %	대입	= *= /= += -= &= ^=  = <<= >>= >>>=
증감	++ --	비교	> < >= <= == !=
비트	&   ^ ~	논리	&&    !
시프트	>> << >>>	조건	? :

## 산술 연산자

- 5 가지 : 더하기(+), 빼기(-), 곱하기(\*), 나누기(/), 나머지(%)

```
var x = 32;  
var total = 100 + x*2/4 - 3; // total은 113
```

- 연산의 결과는 항상 실수

```
var div = 32/10; // div = 3.2
```

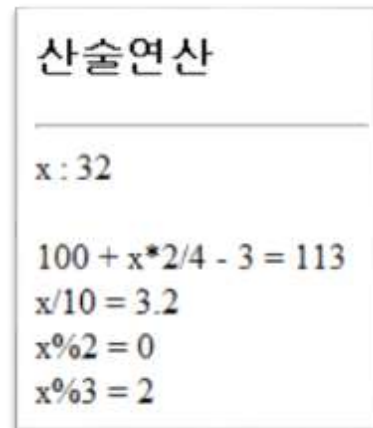
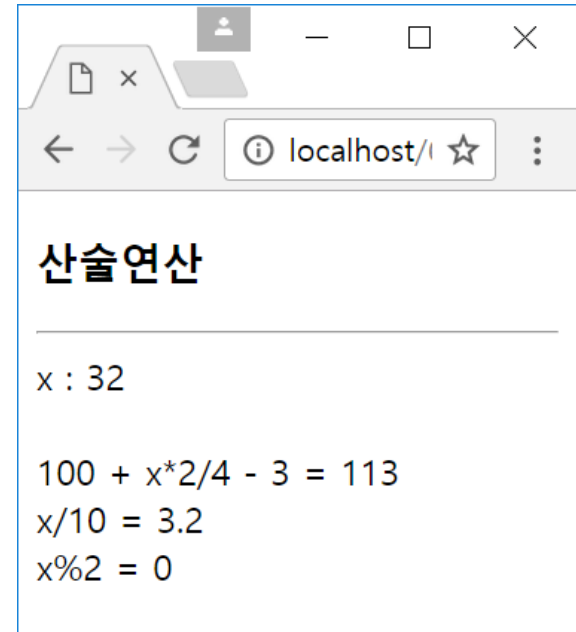
# 예제 6-8 산술 연산

43

```
<!DOCTYPE html>
<html>
<head> <title>산술연산</title> </head>
<body>
<h3>산술연산</h3>
<hr>
<script>
  var x=32;
  var total = 100 + x*2/4 - 3; // total은 113
  var div = x / 10; // div는 3.2
  var mod = x % 2; // x를 2로 나눈 나머지, 0
  var mod2 = x % 3;

  document.write("x : " + x + "<br><br>");
  document.write("100 + x*2/4 - 3 = " + total + "<br>");
  document.write("x/10 = " + div + "<br>");
  document.write("x%2 = " + mod + "<br>");
  document.write("x%3 = " + mod2 + "<br>");
</script>
</body>
</html>
```

wk09js\_ex08\_operation.html

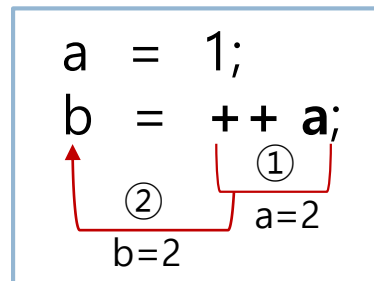


# 증감 연산자

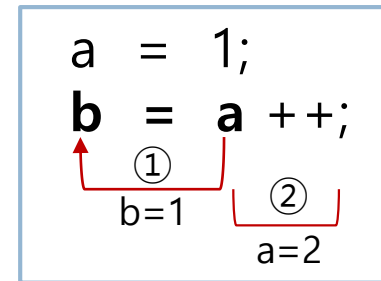
44

## 증감 연산자 : ++, --

(a) 전위연산자



(b) 후위연산자



연산자	내용	연산자	내용
a++	a를 사용한 후에 a 값 1 증가	++a	a를 1 증가한 후에 a 값 사용
a--	a를 사용한 후에 a 값 1 감소	--a	a를 1 감소한 후에 a 값 사용



# 대입 연산자

45

- 대입 연산 : 오른쪽 식의 결과를 왼쪽에 있는 변수에 대입

```
var a=1, b=3;  
a = b;    // a에 b의 값이 대입되어 a=3, b=3이 된다.  
a += b;   // a = a + b의 연산이 이루어져, a=6, b=3이 된다.
```

- 대입연산자 종류

연산자	내용	연산자	내용
a = b	b 값을 a에 대입	a &= b	a = a & b와 동일
a += b	a = a + b와 동일	a ^= b	a = a ^ b와 동일
a -= b	a = a - b와 동일	a  = b	a = a   b와 동일
a *= b	a = a * b와 동일	a <<= b	a = a << b와 동일
a /= b	a = a / b와 동일	a >>= b	a = a >> b와 동일
a %= b	a = a % b와 동일	a >>>= b	a = a >>> b와 동일

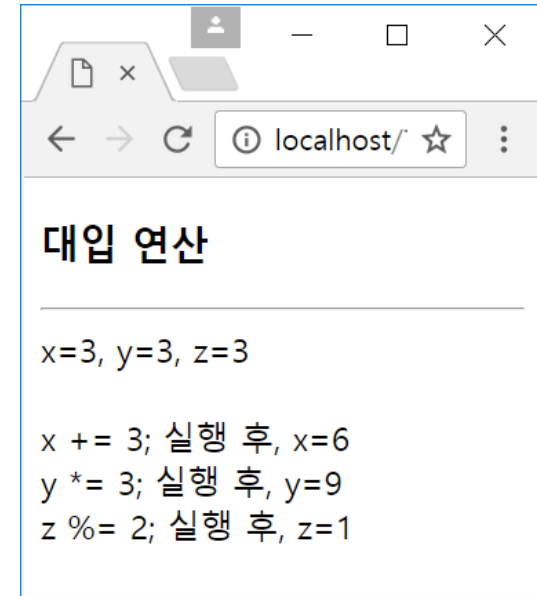
# 예제 6-9 대입 연산

46

```
<!DOCTYPE html>
<html>
<head> <title>대입 연산</title> </head>
<body>
<h3>대입 연산</h3>
<hr>
<script>
  var x=3, y=3, z=3;
  document.write("x=" + x + ", y=" + y);
  document.write(", z=" + z + "<br><br>");

  x += 3; // x=x+3 -> x=6
  y *= 3; // y=y*3 -> y=9
  z %= 2; // z=z%2 -> z=1

  document.write("x += 3; 실행 후, x=" + x + "<br>");
  document.write("y *= 3; 실행 후, y=" + y + "<br>");
  document.write("z %= 2; 실행 후, z=" + z);
</script>
</body>
</html>
```



# 예제 6-9A 복잡한 연산 (추가, exam)

47

```
<!DOCTYPE html>
<html>
<head> <title>대입 연산</title> </head>
<body>
<h3>대입 연산 II</h3>
<hr>
<script>
  var x=3, y=3, z=3;
  document.write("x=" + ++x + ", y=" + y-- + "<br>");
  document.write("x=" + x + ", y=" + y);
  document.write(", z=" + z + "<br><br>");

  x = x-- + 3; //
  y = ++y * 3; //
  z = (++z) % 2; //

  document.write(" x = x-- + 3; 실행 후, x=" + x + "<br>");
  document.write(" y = ++y * 3; 실행 후, y=" + y + "<br>");
  document.write(" z = (++z) % 2; 실행 후, z=" + z);
</script>
</body>
</html>
```

## 대입 연산 II

x=4, y=3

x=4, y=2, z=3

x = x-- + 3; 실행 후, x=7

y = ++y \* 3; 실행 후, y=9

z = (++z) % 2; 실행 후, z=0

# 비교 연산자

48

- 비교 연산 : 두 값 비교, true나 false의 결과를 내는 연산

```
var age = 25;  
var result = (age > 20); // age가 20보다 크므로 result는 true
```

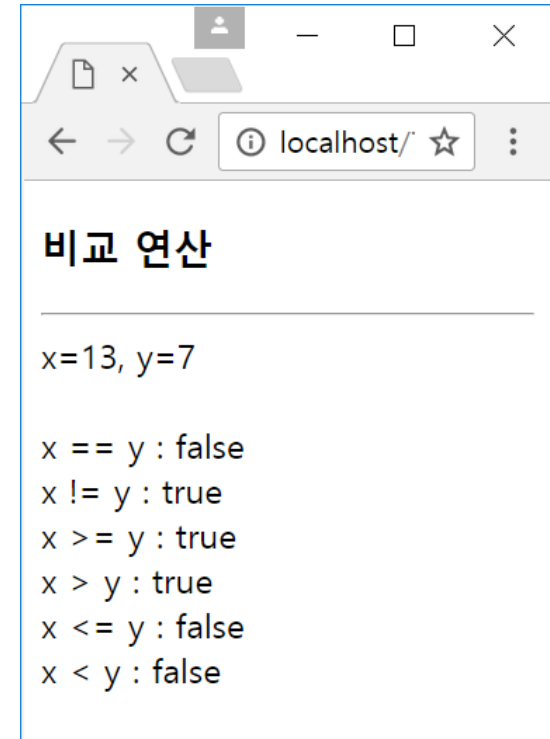
- 비교 연산자 종류

연산자	내용	연산자	내용
$a < b$	a가 b보다 작으면 true	$a \geq b$	a가 b보다 크거나 같으면 true
$a > b$	a가 b보다 크면 true	$a == b$	a가 b와 같으면 true
$a \leq b$	a가 b보다 작거나 같으면 true	$a != b$	a가 b와 같지 않으면 true

# 예제 6-10 비교 연산

49

```
<!DOCTYPE html>
<html>
<head> <title>비교 연산</title> </head>
<body>
<h3>비교 연산</h3>
<hr>
<script>
  var x=13, y=7;
  document.write("x=" + x + ", y=" + y + "<br><br>");
  document.write("x == y : " + (x == y) + "<br>");
  document.write("x != y : " + (x != y) + "<br>");
  document.write("x >= y : " + (x >= y) + "<br>");
  document.write("x > y : " + (x > y) + "<br>");
  document.write("x <= y : " + (x <= y) + "<br>");
  document.write("x < y : " + (x < y) + "<br>");
</script>
</body>
</html>
```



wk09js\_ex10\_comparison.html

# 논리 연산자

50

## □ 논리 연산 : AND, OR, NOT

```
var score = 90;  
var age = 20;  
var res = ((score > 80) && (age < 25)); // res=true
```

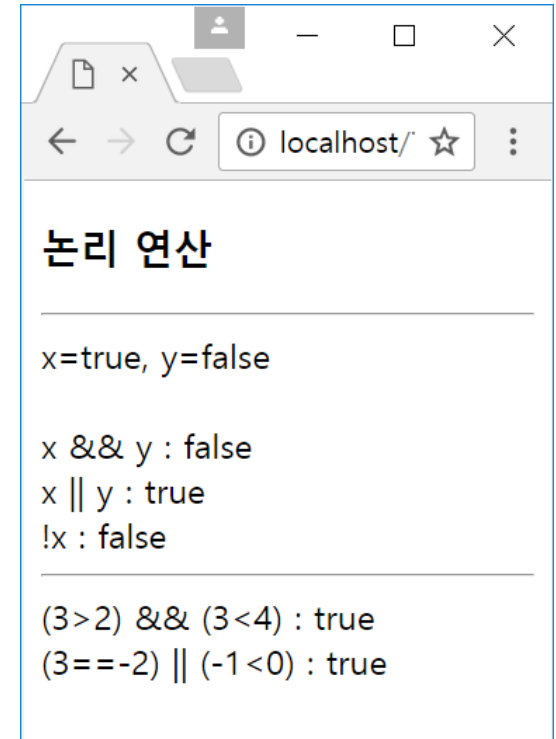
## □ 논리 연산 종류

연산자	별칭	내용
a && b	논리 AND 연산	a, b 모두 true일 때 true 리턴
a    b	논리 OR 연산	a, b 중 하나라도 true이면 true 리턴
!a	논리 NOT 연산	a가 true이면 false 값을, false이면 true 값 리턴

# 예제 6-11 논리 연산

51

```
<!DOCTYPE html>
<html>
<head><title>논리 연산</title></head>
<body>
<h3>논리 연산</h3>
<hr>
<script>
  var x=true, y=false;
  document.write("x=" + x + ", y=" + y + "<br><br>");
  document.write("x && y : " + (x&&y) + "<br>");
  document.write("x || y : " + (x||y) + "<br>");
  document.write("!x : " + (!x) + "<br>");
  document.write("<hr>");
  document.write("(3>2) && (3<4) : " + ((3>2)&&(3<4)) + "<br>");
  document.write("(3== -2) || (-1<0) : " + ((3== -2)||(-1<0)));
</script>
</body>
</html>
```



wk09js\_ex11\_logical.html

# 조건 연산자

52

## □ 조건 연산 (삼항연산)

▣ `condition ? expTrue : expFalse`

- `condition`이 `true`이면 전체 결과는 `expTrue`의 계산 값
- `false`이면 `expFalse`의 계산 값

```
var x=5, y=3;
```

```
var big = (x>y) ? x : y; // (x>y)가 true이므로 x 값 5가 big에 대입된다.
```

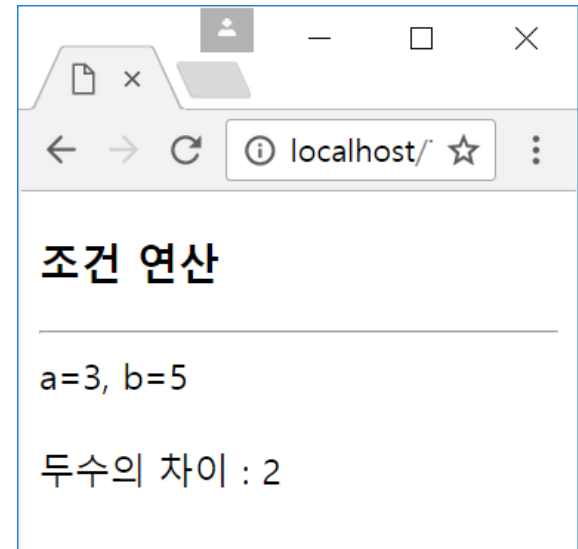


# 예제 6-12 조건 연산

53

```
<!DOCTYPE html>
<html>
<head> <title>조건 연산</title> </head>
<body>
<h3>조건 연산</h3>
<hr>
<script>
  var a=3, b=5;
  document.write("a=" + a + ", b=" + b + "<br><br>");
  document.write("두수의 차이 : " + ((a>b)?(a-b):(b-a)));
</script>
</body>
</html>
```

wk09js\_ex12\_conditional.html



# JS-Project I: simple calculator

54

## 간단한 계산기

첫번째 정수:

두번째 정수:

계산 결과 :



# JS-Project 1.1: 간단한 덧셈

55

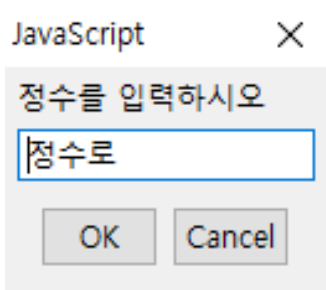
```
<script>
  var x, y;
  var input;

  input = prompt("정수를 입력하십시오", "정수로");
  x = parseInt(input);

  input = prompt("정수를 입력하십시오", "정수로");
  y = parseInt(input);

  document.write(x + y + "<br>");
</script>
```

hmxx\_calc\_01.html



$x + y = 6$

**JS App: Simple calculator**

간단한 계산기



**JS App: Simple calculator**

간단한 계산기

$x + y = 6$

# Project 1.2: 덧셈 계산기 (calc() 사용)

56

```
<body>
  <h3>덧셈 계산기</h3>
  <form name="myform" action="..." method="post">
    첫번째 정수:
    <input id="x" /><br />
    두번째 정수:
    <input id="y" /><br />
    합계:
    <input id="sum" /><br />
    <input type="button" value="계산" onclick="calc();" />
  </form>
</body>
</html>
```

hmxx\_calc\_02.html

**JS App: Simple calculator**

---

덧셈 계산기

첫번째 정수:

두번째 정수:

합계:

# JS-Project 1.3: **calc()** using **getElementById()**

57

```
<html>
<head>

  <title>Javascript calculator</title>
  <script>
    function calc() {

      var x = document.getElementById("x").value;
      var y = document.getElementById("y").value;
      var result;

      result = parseInt(x) + parseInt(y);
      document.getElementById("sum").value = result;
    }
  </script>
</head>
```

hmxx\_calc\_02.html

# JS-Project 1.4: 간단한 계산기 (+, -, \*, /, %)

58

## 간단한 계산기

첫번째 정수:

두번째 정수:

계산 결과 :

## 간단한 계산기

첫번째 정수:

두번째 정수:

계산 결과 :

## 간단한 계산기

첫번째 정수:

두번째 정수:

계산 결과 :

# JS-Project 1.5: 간단한 계산기 (+, -, \*, /, %)

59

```
<body>
  <h3>간단한 계산기</h3>
  <form name="myform" action="..." method="post">
    첫번째 정수:
    <input id="x" /><br />
    두번째 정수:
    <input id="y" /><br />
    계산결과:
    <input id="result" /><br />
    <input type="button" value="+" onclick="calc_add();" />
    <input type="button" value="-" onclick="calc_sub();" />
    <input type="button" value="*" onclick="calc_mul();" />
    <input type="button" value="/" onclick="calc_div();" />
    <input type="button" value="%" onclick="calc_mod();" />
  </form>

</body>
</html>
```

## JS App: Simple calculator

### 간단한 계산기

첫번째 정수:

두번째 정수:

계산결과:

# JS-Project 1.6: 간단한 계산기 (+, -, \*, /, %)

60

```
function calc_add() { // 덧셈

    var x = document.getElementById("x").value;
    var y = document.getElementById("y").value;
    var result;
    result = parseInt(x) + parseInt(y);
    document.getElementById("result").value = result;
}

function calc_sub() { // 뺄셈
    var x = document.getElementById("x").value;
    var y = document.getElementById("y").value;
    var result;
    result = parseInt(x) - parseInt(y);
    document.getElementById("result").value = result;
}
```

hmx\_x\_calc\_03.html

## JS App: Simple calculator

### 간단한 계산기

첫번째 정수:

두번째 정수:

계산결과:



# JS-project-I. hmxx\_calc.html

61

[JS project I.] 자바스크립트를 이용하는 간단한 계산기를 완성하시오.

- ① 계산기 - 다음 연산 구현 : add(+), subtract(-), **multiply(\*), divide(/), mod(%)**
- ② **파일명: hmxx\_calc.html**
- ③ **가점: Javascript 프로그래밍 응용 능력.**

# wk09-실습 : 결과를 나의 github에 올리기

62

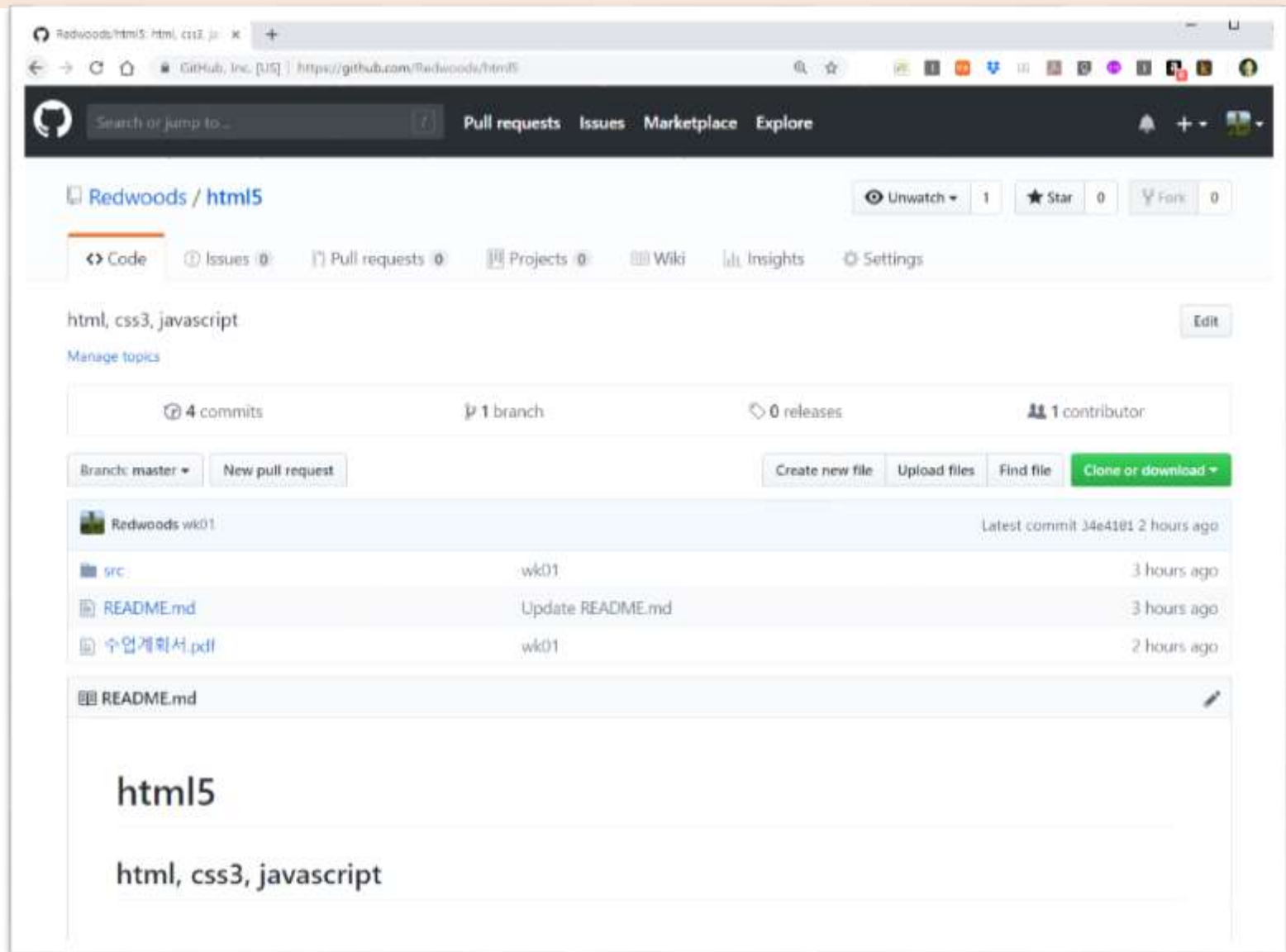
실습 결과를 github에 올립니다.

1. README.md에는 실습 결과 요약 추가 입력
2. hmxx\_calc.html 완성
3. "hmxx" repo 에 wk09 폴더 upload

단 업로드가 안될 경우, wk09.zip을 업로드  
그리고 집에서 wk09 폴더로 다시 업로드.

# HTML5 : 강의자료실

## <https://github.com/redwoods/html5>



# 교재 WEB 강의 소개

← → ↻ ① webprogramming.co.kr ☆



HTML5+  
CSS3+  
Javascript

웹 프로그래밍

Home Introduction Notice Board Support Code



HTML5 +  
CSS3 +  
Javascript

웹 프로그래밍

HTML5로  
여러분의 무한한  
상상력을  
표현해 보세요!



Sir Tim Berners-Lee  
(1955.6.8 ~)

명품 웹 프로그래밍 소개  
"웹 프로그래밍을  
가장 쉽게 익힐 수 있는 책"



처음 웹 프로그래밍을 공부하는 입문자들도  
모든 주제를 직관적으로 이해하고  
빠르게 파악할 수 있습니다.

자세히보기 →

강력한 Q&A 피드백 제공  
"빠르고, 간결하고, 정확한  
저자의 직접적인 답변"



'이거 이해가 잘 안되는데.. 물어볼 사람도 없고..'  
데마상 고민하지 마세요.  
명품 웹 프로그래밍 홈페이지에서는  
누구나 저자가 직접 답변해주는  
Q&A 게시판들을 이용하실 수 있습니다.

자세히보기 →

즉석 실행 가능한 예제 프로그램  
"백문이 불여일견, 백견이 불여일타(打)!"



코드뿐만 실행되어 있는 예제들.  
결과 화면이 있어도 이해가 잘 안되시죠?  
예제 소스를 바탕으로, 내림대로 수정한  
코드를 즉석으로 웹 페이지로  
변환해주는 예제 프로그램을 통해  
모든 코드를 빠르고 쉽게  
이해할 수 있습니다.

자세히보기 →

Notice

Test

2017-01-16 15:32

Know-How

Test

2017-01-17 14:04 관리자

64

<http://webprogramming.co.kr>

# 관련 WEB 강의 소개 – w3schools.com

The screenshot shows the w3schools.com website. The browser address bar displays "https://www.w3schools.com". The website has a green header with the logo "w3schools.com" and the tagline "THE WORLD'S LARGEST WEB DEVELOPER SITE". Below the header is a navigation bar with "TUTORIALS", "REFERENCES", and "EXAMPLES". On the left side, there is a sidebar menu listing various topics: HTML and CSS, JavaScript, Server Side, Web Building, and XML Tutorials. The main content area is divided into three sections: HTML, CSS, and JavaScript. Each section has a title, a subtitle, and a "Try It Yourself" button. The HTML section includes a code example for a basic HTML document. The CSS section includes a code example for styling a body and a div. The JavaScript section includes a code example for a function that gets an element by ID and changes its style.

HTML and CSS

- Learn HTML
- Learn CSS
- Learn W3.CSS
- Learn Colors
- Learn Bootstrap
- Learn Icons
- Learn Graphics
- Learn How To

JavaScript

- Learn JavaScript
- Learn W3.JS
- Learn JQuery
- Learn JQueryMobile
- Learn AppML
- Learn AngularJS
- Learn JSON
- Learn AJAX

Server Side

- Learn SQL
- Learn PHP
- Learn ASP

Web Building

- Web Templates
- Web Statistics
- Web Certificates

XML Tutorials

- Learn XML
- Learn XML AJAX
- Learn XML DOM
- Learn XML DTD
- Learn XML Schema
- Learn XSLT
- Learn XPath
- Learn XQuery

## HTML

The language for building web pages

LEARN HTML HTML REFERENCE

HTML Example:

```
<!DOCTYPE html>
<html>
<title>HTML Tutorial</title>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Try It Yourself »

## CSS

The language for styling web pages

LEARN CSS CSS REFERENCE

CSS Example:

```
body {
  background-color: lightblue;
}
div {
  color: white;
  text-align: center;
}
p {
  font-family: verdana;
  font-size: 28px;
}
```

Try It Yourself »

## JavaScript

The language for programming web pages

JavaScript Example:

```
<script>
function myFunction() {
  var x = document.getElementById("demo");
  x.style.backgroundColor = "red";
}
```